**1A2B (Bulls & Cows) Game Design/Development**

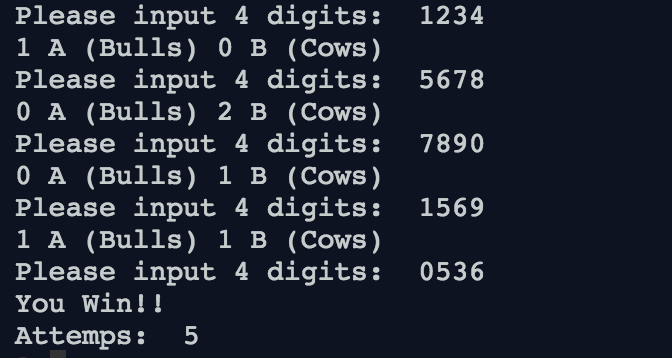
We are about to develop the game, called 1A2B (aka. Bulls & Cows) in Python. The game is played as follows:

1. In the beginning, the game will randomly pick four different digits as the target number (e.g., 2468) to be guessed by the player.
2. In each round of the game, the player can guess four digits, e.g., 1234. If an user-guessed digit hits the target number and its position is the same as that in the target number, it is considered as 1A (or 1 bull). If an user-guessed digit hits the target number but its position is different from that in the target number, it is considered as 1B (or 1 cow).

For example, if the target number is 2468 and the user guessing is 1234, the result of this round is 2B because both digits 2 and 4 hit the target but their positions are wrong. If the user guessing is 2478, the result of this round is 3A because both digits 2, 4, and 8 hit the target number, and their positions are all correct.

1. The player wins the game if his/her guessing results in 4A (or 4 bulls), i.e., all the digits appear in the target number, and their positions are correct.

The screen shot of the game could be as follows:



More information about the game: https://en.wikipedia.org/wiki/Bulls\_and\_Cows

**Discussion 1**

To play the 1A2B game, what are the steps (procedures) needed? What is the flow of them? Come up with your design (algorithm) by pseudocode/flowchart first.

1.Select a target number of 4 digits

2.Get user input of 4 digits

3. Compare user input and target no. If number at correct position, A+=1. If number at different positions, B+=1. Print the A and B no.

4.If guess wrong go back to step 2

5.If 4A, print you win

6.Count the number of attempts

**Discussion 2**

Based on your design, define the prototype of functions corresponding to each step.

* What is the functionality?
* How many parameters? What are they?
* The return object of the function

Def getTnum(Tnum,Gnum):

Return Tnum

Def getGuess(guess):

Return guess

Def compare(guess,Tnum):

Return A

…

Return B

**Def gennumbers(numlist,count):**

**Def userguess():**

**Def checkguess(guesslist,answer)**

**Discussion 3**

Implement each function according to your design.

Def getTnum(Tnum,Gnum):

Gnum = [str(r.randint(0,9)), str(r.randint(0,9)), str(r.randint(0,9)), str(r.randint(0,9))]  
 Tnum = "".join(Gnum)

Return Tnum

Def getGuess(guess):  
 guess = ""  
 while guess.isdigit() == False or len(guess) != 4:  
 guess = input("Enter a 4 digit number: ")

Return guess

Def compare(guess,Tnum):

for i in range(len(Tnum)):  
 if Tnum[i] == guess[i]:  
 A += 1

return A  
 elif Tnum[i] in guess:  
 B += 1

return B

import random

numlist = [0,1,2,3,4,5,6,7,8,9]

def gennumbers(numlist,count):

random.shuffle(numlist)

numbers = []

for I in range(count):

numbers.append(numlist[i])

return numbers

def userguess():

inputstr = input(“input…”)

while input str.isdigit() == False or len(inputstr) != WIDTH:

inputstr = input(“Wrong format”)

guess=[]

for I in range(WIDTH)

num = int(inputstr[I])

guess.append(num)

Return guess

Def checkguess(guesslist, answerlist):

Bulls = 0

Cows = 0

For I in range(len(guesslist)):

For j in range(len(answerlist)):

If guesslist[i] ==answerlist[j]:

If I ==j:

Bulls+=i

Else:

Cows+=i

Return(bulls,cows)

**Discussion 4**

Combine all functions you developed to complete the game.

import random  
r = random  
  
Gnum = [str(r.randint(0,9)),str(r.randint(0,9)),str(r.randint(0,9)),str(r.randint(0,9))]  
Tnum = "".join(Gnum)  
print(Tnum)  
  
A = 0  
B = 0  
  
while A != 4:  
 A=0  
 B=0  
 guess = ""  
 while guess.isdigit() == False or len(guess) != 4:  
 guess = input("Enter a 4 digit number: ")  
  
# count position and number  
 for i in range(len(Tnum)):  
 if Tnum[i] == guess[i]:  
 A += 1  
 elif Tnum[i] in guess:  
 B += 1  
  
 print(A)  
 print(B)

answer = gennumbers(numlist, width)

attempts = 0

while True:

guesses = userguess()

attempts +=1

result = checkguess(guesses,answer)

if result[0] == WIDTH :

print(“you win”)

print(“Attempts”,attempt)

break

else:

print(result[0],…)